SetLink: the CERN Document Server Link Manager

Jean-Yves Le Meur; David McGlashan
15/01/2000

Published in HEP Libraries Webzine, Issue 1
CERN-ETT-2000-001

This paper analyses the problems encountered by electronic libraries to cope with long term keeping of on-line documents. What is the best solution not to fill in a library system with non persistent addresses ? Different possible solutions are studied, based on the experience of the CERN Document Server and its library.

This paper will stress the importance of using a Link Manager for any long term Web server. It will explain how the CERN SetLink Link Manager is designed to handle a wide range of document types and formats, from photos in JPEG to eprints in PDF. It will also focus on other possibilities offered by using such an application, like automatic figures extraction or concatenation, full text searching and on-the-fly format conversions.

Background: the CERN library and Document Server

In order to scale the problem, this first part shortly describes an example of an average size library with a huge amount of electronic resources: the CERN library. As of 15th February 2000 this database contains more than 380.000 documents in the form of bibliographic notices and more than 140.000 documents are available in an electronic format for full text download.

A large repository of meta-data ... and of data

CERN is the European Organization for Nuclear Research [1], situated in Switzerland, near to Geneva. It is CERN policy to publish many of its findings so the library keeps huge numbers of preprints, books, periodicals, conference reports and High Energy Physics institute papers amongst other things.

This vast repository of meta-data contains more than 255 000 preprints, 43 000 books, 1 200 periodicals, 9 000 administrative documents, 25 000 official archived records, 2 050 photos, 1500 press cuttings, etc., and it is always growing. The description of a document (its fields) is achieved using a customization of the very complete MARC [2] format, stored in the Aleph automated library system [3].

The CERN system is also composed of a large quantity of data: electronic versions of full text documents (PS and PDF), paper versions which are scanned to be delivered electronically, eprints, scientific committee papers, academic training, photos, press cuttings, etc. The CERN Document Server stores today about 70 000 files, which represents about half of the total fulltext linking done from the Library system.

Two types of URL

All of the library is available through the Web, using WebLib [4] which is a search engine built on top of the meta-data. In order to enable users to access full text of documents (if available), a URL must be provided so that the search finally leads to its final objective: read the document !

As any Internet user will confirm, the trouble with URL is its very short lifetime. The so-called "U-N-F" - URL Not Found - problem can result from changes at three levels: the protocol (ftp -> http -> https) which is rare, the server name (Error 602: Connection failed) which is more often and the file names (the famous Error 404) which is so frequent.

In order to cope with this problem within the library system, we distinguish two types of URL, the stored ones and the derived ones.

- **Stored URL**: specific fields do exist in the database where complete addresses are kept.
  They are considered as *controlled URL* if they point to the CERN document server or to some official CERN site (which will be maintained for the lifetime of CERN).
  They are *uncontrolled URL* if they link to documents or pages outside the laboratory, or possibly to "wild" CERN servers. Of course, this last type of URL are the most problematic. In order to isolate broken links, the only solution is to run an URL Checker like MOMspider [5] which is run regularly until it can certify the address is broken. Correction or deletion must then be done "manually", one by one. The first time it was run on the book collection (summer 1998), already half of the URL were obsolete.
  In general, the stored URL are easy to collect and to check but their maintenance is a real pain, even with the help of software.

- **Derived URL**: these are addresses which are created on-the-fly from another information in the meta-data set.
  It can be derived from a report number, from a publication reference, from a conference code or from any other key value. The link is generated by the program displaying the results of a search. The main trouble encountered with this type of URL is that broken addresses are generally discovered by chance. No URL checkers can be run on "on-the-fly" URL. If an electronic journal changes the organization of its articles URL, or if it disappear, all the links to it become dead links. The advantage of this technology is that a single maintenance action (e.g.: remove the e-journal name in the list of CERN e-jounals on-line) will be understood by the program, and all the linking remains valid. Sometimes, it may be more complex (e.g.: introducing issue numbers in URL to articles) and it then requires some maintenance at the programming level. Still, only one maintenance effort may correct thousands of links.
  So, even if it is a bit problematic to check the persistency of derived URL, they are much easier to maintain than uncontrolled stored URL.

General solutions to broken links

We are not going into the details of the various solutions proposed to solve the global Addressing Internet problem [6]. Let us only mention the long-term solution of URN, Uniform Resource Name, which is a generic name for many possible physical names. The basic idea is that Domain name servers

will be able to handle URNs in order to find out correct URL (locators).

Meanwhile, resolution services should exist on the Web servers themselves. Links to files should be systematically replaced by links to a resolution application (a link manager) in charge of fetching the requested page. Examples of such a technology are PURL (Persistent URL system) [7] or DOI (Digital Object Identifier) [8].

As the CERN library system is not only a "URL consumer" but also a "URL provider" - many sites are pointing to library resources or to full text documents - The CERN Document Server has developed its own link manager (called **SetLink**) which serves today the 70 000 full text documents and photos. In this way, all links from the library databases to the CERN Document Server will be long term: independent from server names, files locations or files formats.

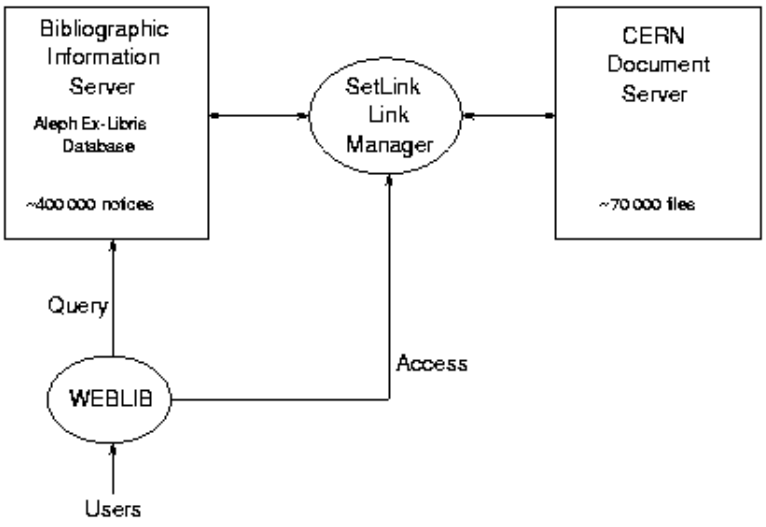
We will see in the next chapter what is the SetLink link manager and why it was decided to use an in-house link manager instead of a commercial alternative.

The SetLink Link Manager

Each record in the library databases, whether its a photograph, a preprint, or whatever, can be found using three simple string parameters: base, category and ID. The base defines the type of document you are looking for, "preprint" for preprints, "PHO" for photographs, etc. Category divides the base into sections, maybe by class or by year, and ID provides a simple identifier for each record within the class, often this is simply an 8 digit number, but it can be a more complex string. SetLink uses this same system to retrieve the files over the web in order to provide consistency and stability in-between the ALEPH database and the CERN Document Server.

SetLink is not a search engine, but for a given base, category and ID combination it should be able to return the document the user is searching for, and offer some additional services in return. You can, for example, convert the document files to other file formats such as PDF, PostScript or even into a GIF image page by page, perform keyword searches on PDF documents, or send the document to a local printer or to your email account, all from the same web page returned by SetLink.

Overview of the structure of WebLib, SetLink and CDS:



Users query the meta-data stored into a bibliographic information database and access the full text documents via the link manager.

A Commercial Alternative

Instead of programming a new link manager for themselves, one alternative solution for the Library Support might have been to investigate several commercial Link Managers for suitability for the task. There are several available, such as Periwinkle [9] or netImpact [10], but these seemed mostly to cater toward organizing hyperlinks within a site rather than dealing with database records such as SetLink does.

Other factors also needed to be considered, such as cost and functionality. It would be cheaper for Library Support to have developed SetLink for themselves, rather than to pay for a license to use a commercial Link Manager. SetLink would also function exactly as is required, and be much easier to maintain and customize if it were developed in-house.

SetLink also performs many "under the bonnet" functions which the end user is not aware of, such as making a record of links which are broken, documents which can be converted to other formats, as well as sending email to the Library Support team when other things go wrong. No existing Link Manager was available (in 1998) which would have performed some of the very specific functions that SetLink does.

A Little History Lesson

In 1994 the CERN Document Server was known as the CERN Preprint Server, and it stored only a fraction of the current library. Only one catalogue existed; the Preprints. There was no SetLink or WebLib at this time, and the full text access was provided by an API running of the Library server.

In 1996 WebLib was introduced on the Preprint server along with the first version of SetLink. This was

a simple version of the program which offered the file for downloading, and a few simple file format conversions.

In 1998 SetLink was upgraded to version 2, which offered a lot more features as well as file format conversions. The differences between versions 1 and 2 were considerable as version 2 represented a complete system redesign and rewrite of the source code.

SetLink was upgraded further in 1999, though the changes made at this time were mostly cosmetic. All the HTML written by the program was tidied up and a cascading style sheet was implemented. The program was also optimized and made to run at a faster speed. Today, 15 different types of documents (bases) are handled by SetLink and 102 possible categories are declared.

Technical Description of SetLink

In the design stage of SetLink version 2, it was decided that the program should be modular, consisting of many parts rather than one complete program. This problem was addressed on two fronts. Firstly, a configuration file for SetLink 2 was designed and created, which meant that one could specify how SetLink would behave in a separate file which would be easy to read. This also brought in the benefit of being able to alter how the program behaves quickly and easily, without having to recompile and then debug the program.

Secondly, the way in which SetLink writes all its HTML back to the browser was altered. Instead of having lines of code in the program to write generic HTML to the browser, the HTML content was removed and divided into smaller files which were fragments of a larger page. The program would select a fragment file for each part of the web page and parse it instead.

SetLink configuration file

The original SetLink was essentially dictated by the same algorithm for every base across the board. It would analyze its parameters, work out a directory on the local server where the files should be and search for them there. If it found them it created hyperlinks for them, otherwise it displayed a simple message saying that the files were unavailable.

The major difference in how SetLink behaved each time was dictated by the "base" parameter passed to it at run-time. This parameter dictates the content of the HTML written by the program, what methods the program uses to look for the files on the document server, which services were offered etc.

The configuration file was designed from the analysis of the differences between each base. It is laid out in terms of each base, with a series of parameters to define that base’s behavior. When the SetLink program runs each time it checks the base parameter and draws its behavior from the parameters specified in the configuration file.

In addition to the base definitions the configuration file also holds useful information about the local file system, such as the software commands for performing the file format conversions, and the locations of other useful files.

Example base definition from the SetLink v2.1 configuration file:

```
// Special case for EXT
//
base(preprint) {
   directory: /archive/electronic/cern/preprints
    category: ext
 shPrefixers: |/$p|/$m5i4|/../scan/$m5i4
     formats: pdf|ps.gz|ps
    bodyHTML: validHtml/pdf/body.html
   linksHTML: validHtml/preprint/links.html
    tailHTML: validHtml/cds/tail.html
   wildcards: _*|fig
 noCategHTML:
    noIDHTML: validHtml/cds/notAvailable.html
  noIDScript: /users/setlink/preprint.notFound.sh $1 $2 $3 $D
}
```

SetLink's HTML

The original SetLink script used to write HTML to the browser directly from the source code. This made the program code about 40% bigger as roughly every third line was writing HTML. In order to make SetLink more modular all of these lines were removed and the HTML content was put into smaller separate files, which are parsed by the program at run-time.

Typically these HTML fragments broke up the page into sections: header, body, links and tail. The header and tail fragments are mostly the same for each base across the board. The main differences between bases occur in the body and links fragments, for example the body and links fragments associated with the Photographs base have <IMG> tags included to display thumbnails of the photos in the database on the SetLink page.

In order that the HTML be interactive with the program a few rudimentary codes were inserted into the HTML, now known as dollar commands because they are preceded with a "$" character. At first these were reasonably simple things such as $Fn for a filename. Hyperlinks to a document file are created thus;

You can download this document in the following formats:
<ul>
<li>Portable Document Format
*
*
Other simple dollar commands were $Fs for the size of the file in kilobytes, $1 for the base, $2 for the category and $3 for the id parameters. This was fine for the start, but things started to become complex when SetLink had to list the contents of a sub directory, or mask a section of HTML which didn't apply to this particular file.
In the end the $ commands became like a programming language in their own right, with a boolean logic of IF/ELSE statements which could be nested up to eight levels deep, and a whole slew of over forty other commands for file information, listing types of file or directories and offering services.

Although 40% of the source code for the original script had been removed by not including any HTML writing lines, they were replaced instead with a considerable chunk of code to parse the HTML and interpret and perform the $ commands, which amounts to around 25% of SetLink 2.1's 2,500 lines.

Very different examples of what SetLink can render can be seen at the URLs below:

Photos: http://documents.cern.ch/SetLink?base=PHO&categ=photo-ac&id=9912012
CERN Yellow Report:
http://documents.cern.ch/SetLink?base=cernrep&categ=Yellow_Report&id=99-07
HEP paper: http://documents.cern.ch/SetLink?base=preprint&categ=ext&id=EXT-99-037

Valid HTML and Style sheets

This design of SetLink consisted of all valid HTML and the use of a cascading style sheet to make the pages appear uniform. A menu bar was placed at the top of the pages, and a tool-bar to assist in using the application was added to every page on the left hand side.
The introduction of the valid HTML means that the pages SetLink creates are viewable on all platforms, and with any browser, and also that they will remain readable until such a time that the web has evolved beyond HTML, and it is no longer recognized by the browsers.
Valid HTML means that the source HTML is written properly, and is checked by a script for errors. Every SetLink page carries a small icon at the bottom to run a checking script available on the pages of the World Web Consortium [11].

The cascading style sheet was introduced as a method of controlling the presentation of the Library Support section's web pages easily, and also in response to industry trends towards using them. The style sheet controls how each page looks in terms of color, fonts and some measurements. This is defined in a separate file which all the library's web pages reference.

Using the style sheet is beneficial for the user who can override our style sheet definition with their own. This is of particular interest to disabled users who can make our pages more readable using a style sheet more suited to their own disability.

Limitations and Perspectives

File Organization

One potential fault with SetLink comes from the organization of files on the CERN Document Server. There are many different types of files, and many different methods of organizing them which exist on the server. Some of these are better than others, obviously, but it is the sheer volume of different methods in use on the server that causes SetLink a headache. It has to be versatile enough to work in ALL cases, without fail. To say that a file does not exist on the Document server, when in fact it does is unacceptable.

The system SetLink uses to extrapolate a sub directory name is a somewhat complex and bewildering notation for editing strings on the fly, and while it is fairly powerful it cannot cope with every eventuality. In many cases files are stored in sub directories beneath the usual base/categ format, and whilst the name of the directory can often be extrapolated from the id parameter, instructing SetLink to do this can be quite tricky, especially in a modular fashion.

External factors

SetLink has no control over the external links it offers from the web pages it creates. These links are to other sites elsewhere on the web. While URL to institutional electronic document archives such as Los Alamos in Santa Fe, New Mexico are reliable, URL left by the author of a document has a quite uncertain retention validity. Although Library Support can guarantee that SetLink URL are permanent, we cannot make such claims about these links to external sites, nor are we responsible for their content.

Another external limitation of SetLink is the poor file format support offered by the current versions of the web browsers. At present, most of the file formats available on the CERN Document Server are not directly viewable through the browsers. It is possible to configure the browser's MIME-types to run the correct application, and there are also plug-ins available to help resolve this problem, but many base level users are unaware of this. For them the only way around the problem is to download the document to their own computer and view it from there, which is a cumbersome solution.

Perspectives

We already know that in the medium term document names, http servers, protocols, storage organization and file formats are likely to change. As an electronic document provider, the CERN Document Server is very keen to push the emergence of XML as a standard format for e-documents. It would enable a massive simplification of the documents handling and a much more effective access for end users.

But whatever is the evolution of the technologies, the current use of the SetLink link manager is a guarantee of a minimal impact on the maintenance of the CERN library system and at the same time the assurance of the ability to quickly link all bibliographic notices to the best corresponding resources.

Finally, we are optimistic that more and more URL providers (like e-journals already involved in Cross References service [12]) will be using link managers in order to decrease the "Error 404" rate and to improve the quality of library services world-wide.

URL References

[1] http://www.cern.ch
[2] http://lcweb.loc.gov/marc/
[3] http://www.exlibris.co.il/
[4] http://weblib.cern.ch/
[5] http://www.ics.uci.edu/WebSoft/MOMspider/WWW94/paper.html
[6] http://www.w3.org/Addressing/
[7] http://purl.oclc.org/OCLC/PURL/INET96
[8] http://www.doi.org/
[9] http://www.periwinkle.net/plinker.html
[10] http://www.netimpact.net/content/link.html
[11] http://www.w3c.org/
[12] http://www.crossref.org/